CLAIMS

1. A computer cluster comprising:

storage media;

a first computer having a first instance of an application program installed, said application program having instructions, said first computer including,

volatile memory;

processing means

for executing instructions of said first instance of said application program so as to modify data stored in said memory,

for creating a snapshot of said data while said first instance of said application program is running, said snapshot being stored in said volatile memory, and

for transferring said snapshot from said volatile memory to said storage media while said first instance of said first instance of said application program is running, and

a second computer having a second instance of said application program installed, said second computer including means for accessing said storage media so that said second instance of said application can access said snapshot as stored on said storage media.

2. A computer cluster as recited in Claim 1 wherein said processing means includes

a data processor

for executing instructions of said first instance of said application program so as to modify data stored in said memory, and

for creating said snapshot of said data while said first instance of said application program is running, said snapshot being stored in said volatile memory, and

a transfer processor for transferring said snapshot from said volatile memory to said storage media while said first instance of said first instance of said application program is running.

3. A computer cluster as recited in Claim 1 further comprising a first cluster daemon running on said first computer for causing said snapshot to be created.

4. A computer cluster as recited in Claim 1 further a second cluster daemon running on said second computer, said second cluster daemon providing:

for detecting a failure that prevents said first instance of said application program from running on said first computer, said failure detector, and

for causing, in response to said detecting a failure, said second computer to process said snapshot in accordance with instructions of said second instance of said application program.

5. A computer cluster as recited in Claim 1 wherein said processing means provides for, in response to a write access of a section of said volatile memory in accordance with instructions of said first instance of said application program, copying data in that section so that one instance of said data originally in that section is modified and the other copy of data originally in that section is not modified.

6. A computer cluster as recited in Claim 2 wherein said data processing means maintains state data, said snapshot data including at least some of said state data.

7. A method comprising:

executing a first instance of an application program on a first computer of a computer cluster so as to generate a series of memory states;

creating a snapshot of one of said states; and

transferring said snapshot to storage media accessible by a second computer of said computer cluster.

8. A method as recited in Claim 7 further comprising executing a second instance of said application program on a second computer of said computer cluster using said snapshot as a starting state.

9. A method as recited in Claim 8 further comprising detecting a failure that prevents execution of said first instance of said application program, said detecting occurring after said transferring and before said executing a second instance.

10. A method as recited in Claim 8 wherein said executing a second instance follows said transferring without an intervening detection of a failure.

11. A method as recited in Claim 7 wherein said transferring is effected by a data transfer processor not used in executing said first instance of said application.

12. A method as recited in Claim 7 wherein said executing is effected by a data processor that stores processor state data internally, said snapshot including said processor state data.